Distributive laws for Lawvere theories

Eugenia Cheng
School of Mathematics and Statistics, University of Sheffield
E-mail: e.cheng@sheffield.ac.uk

December 15, 2011

Abstract

Distributive laws give a way of combining two algebraic structures expressed as monads; in this paper we propose a theory of distributive laws for combining algebraic structures expressed as Lawvere theories. We propose four approaches, involving profunctors, monoidal profunctors, an extension of the free finite-product category 2-monad from **Cat** to **Prof**, and factorisation systems respectively. We exhibit comparison functors between **CAT** and each of these new frameworks to show that the distributive laws between the Lawvere theories correspond in a suitable way to distributive laws between their associated finitary monads. The different but equivalent formulations then provide, between them, a framework conducive to generalisation, but also an explicit description of the composite theories arising from distributive laws.

Contents

Introduction

Lawvere theories were introduced in [9] and were a great breakthrough in the understanding of algebraic theories. They give a different viewpoint from that of monads, as they explicitly implement the notion of arity. One practical advantage of Lawvere theories over monads is that Lawvere theories allow us to study models in different categories, starting from the same Lawvere theory. For example, topological groups and ordinary groups both arise as models for the Lawvere theory for groups, whereas using monads we have to construct a monad on **Set** for groups and a different (albeit related) monad on **Top** for topological groups.

Distributive laws give us a way of combining algebraic theories expressed as monads. The classic example combines the monad for groups and the monad for monoids (both monads being on **Set**) to yield the monad for rings as the "composite" algebraic theory: the distributive law makes the composite of the two monads into a new monad. The theory for combining three or more monads is developed in [3].

It is well-known that Lawvere theories and monads are related—Lawvere theories correspond to *finitary* monads on **Set**. This should not be thought of as a statement that Lawvere theories are "merely" a special case of monads; the above comments about models shows one way in which Lawvere theories are of importance in their own right.

A natural question then arrises—is there a notion of distributive law for Lawvere theories? Of course, given the above correspondence with finitary monads on **Set**, one could simply say "a distributive law for Lawvere theories is a distributive law between the associated finitary monads on **Set**."

However, we seek a formulation that is "native" to the framework of Lawvere theories. In this paper we will provide four equivalent formulations at varying levels of abstraction. As usual we expect the most abstract one to be more useful for theorising, and expect the most concrete one to be more useful for applications.

Our three most abstract formulations will come from observing that Lawvere theories may themselves be thought of as monads inside some other bicategory. Having expressed Lawvere theories in this way it is natural to define distributive laws for Lawvere theories as distributive laws between the monads in these bicategories. The bicategories in question are

1. **Prof**—categories, profunctors and natural transformations.
2. **Prof**(**Mon**)—as above but internal to monoids.
3. $\mathbf{Prof}_{\mathcal{P}}$—the Kleisli bicategory for the free finite-product category 2-monad extended from **Cat** to **Prof**.

The advantage of (1) is that the bicategory **Prof** is well-known and quite easy to understand; however not *all* monads in here are Lawvere theories even if we restrict to the correct underlying 0-cell.

The approach using (2) is in some ways more naturally-arising than (1) and in fact helps us understand it. Also, it is closely related to Lack's work on distributive laws for PROPs [8] and recent developments in [1].

The advantage of (3) is that, once we restrict to the correct underlying 0-cell, *all* monads are Lawvere theories. It is this that enables us to prove that the

composite monad in each of these three frameworks is also a Lawvere theory—it is immediate in (3) and then by the equivalence of the three definitions, the result will follow for (1) and (2).

Another advantage of (3) is that although (or because) this bicategory is very much harder to work with, it affords not only the most precision but also greater flexibility. We will see that monads on other 0-cells may be thought of as "typed" Lawvere theories, and the setting also opens the possibility for changing the 2-monad $\mathcal{P}$ to study different types of theory; this insight is all gained from Hyland [5].

For the most concrete formulation, we unravel (1) and express it in terms of factorisation systems. The notions are equivalent, but the framework feels quite different from the above abstractions and therefore provides different insights. For example, this distributive laws for monads seem suited to considering composition of monads, whereas factorisation systems seem suited to considering *decompositions*.

Note that it is quite easy to make a wrong definition of distributive law for Lawvere theories along the above lines, by working in an ill-chosen bicategory. For example, every Lawvere theory is a monad in **Span** (which is, after all, related to **Prof**), but considering distributive laws in this bicategory gives the wrong notion, as we will show in Section 4.

As evidence that our definitions do give the correct notion, we prove that all our definitions of distributive law for Lawvere theories correspond suitably to distributive laws between the associated monads, with the composite Lawvere theories corresponding to the composite monads. *En passant*, we shed some more, abstract, light on the monad/Lawvere theory correspondence.

The paper is organised as follows. In Section 1 we briefly recall the definition of Lawvere theory and the correspondence with finitary monads on **Set**. In Section 2 we briefly recall the notion of distributive law between monads inside a bicategory. Experts can skip both these sections with impunity. In Sections 3–6 we present our four different approaches to distributive laws for Lawvere theories and in Section 7 we provide the comparison. We finish in Section 8 with some brief comments about the possibilities for future work.

Acknowledgements

This work was launched by a question posed to me by Jean Bénabou at the 89th PSSL in Louvain-la-Neuve, for which I am grateful. Its progress was then dramatically catalysed by invitations I received to speak at the 4th Scottish Category Seminar and at "Category Theory, Algebra and Geometry" in Louvain-la-Neuve in May 2011, and I wish to express my thanks to the organisers of these events, especially Tom Leinster, Marino Gran and Enrico Vitale.

1 Lawvere theories

In this section we recall the basic definitions and results about Lawvere theories that we will need in the rest of this paper. Nothing in this section is new. Lawvere theories were introduced in [9]; we find that [6] gives a useful exposition.

The idea of a Lawvere theory is to encapsulate an algebraic theory in a category $\mathbb{L}$ where

- the objects of $\mathbb{L}$ are the natural numbers, the "arities",
- a morphism $k \longrightarrow 1$ is an operation of arity k, and
- a morphism $k \longrightarrow m$ is m operations of arity k.

Let $\mathbb{F}$ denote a skeleton of **FinSet**, the category of finite sets and all functions between them. So in particular the objects of $\mathbb{F}$ are the natural numbers (including 0).

Definition 1.1. A **Lawvere theory** is a small category $\mathbb{L}$ with (necessarily strictly associative) finite products, equipped with a strict product-preserving identity-on-objects functor

$$\alpha_{\mathbb{L}} \colon \mathbb{F}^{\mathrm{op}} \longrightarrow \mathbb{L}.$$

A **morphism of Lawvere theories** from $\mathbb{L}$ to $\mathbb{L}'$ is a (necessarily strict) finite-product preserving functor making the obvious triangle commute. Lawvere theories and their morphisms form a category **Law**.

Remark 1.2. It is worth making the structure of $\mathbb{F}$ a little more explicit here as we will rely on this heavily later, especially when we consider the free finite-product category monad $\mathcal{P}$ in Section 6. Since **FinSet** is equivalent to the free finite coproduct category on 1, $\mathbb{F}^{\mathrm{op}}$ is equivalent to the free finite product category on 1. Finite products are given by *addition* of natural numbers, and so a morphism

$$\alpha \colon k \longrightarrow m \in \mathcal{P}1$$

is given by, for each $i \in [m]$, a choice of projection $k \longrightarrow 1$. Hence α is precisely a function $[m] \longrightarrow [k]$ where we write $[k]$ for a set of k elements. (We will sometimes omit the square brackets if confusion is unlikely.)

The idea for Lawvere theories is that $\mathbb{F}^{\mathrm{op}}$ encapsulates the operations that must generically exist in any algebraic theory: forgetting and repeating variables. For each $m \in \mathbb{F}^{\mathrm{op}}$ we have:

- the ith product projection $m \longrightarrow 1$ corresponding to forgetting all m variables except the ith one, and
- the diagonal $1 \longrightarrow m$ corresponding to repeating a variable m times.

Definition 1.3. The morphisms in $\mathbb{L}$ are called **operations** and those arising from morphisms of $\mathbb{F}$ are called **basic operations**.

Example 1.4. In the Lawvere theory for monoids, the 2-ary operations, that is, morphisms $2 \longrightarrow 1$, include the operations

$$ab, a, a^2, b, b^2, aba, ab^3a^5, \ldots$$

that is, everything in the free monoid on a 2-element set. This is a different notion of arity from the one used to express algebraic theories via operads—in the operad for monoids the only 2-ary operation is ab.

A morphism $3 \longrightarrow 2$ is given by two 3-ary operations, eg

$$\{abc, ab^2c^2\}, \{bc^2a, ababc\}, \ldots$$

A typical composite looks like

$$3 \xrightarrow{\{abc, ab^2c^2\}} 2 \xrightarrow{\{x^2y\}} 1$$

yielding the composite 3-ary operation $abc \cdot abc \cdot ab^2c^2$.

Note that as a result of forgetting variables we have many different possible arities for the "same" operation. For example starting with a 3-ary operation abc, say, we may precompose with variable-forgetting morphisms to express abc as a k-ary operation where all variables apart from a, b, c are forgotten:

$$\begin{tikzcd}
\cdots \arrow[r, dashed] & 5 \arrow[r, dashed] \arrow[drr] & 4 \arrow[r, dashed] \arrow[dr] & 3 \arrow[d, "abc"] \\
 & & & 1
\end{tikzcd}$$

As a 5-ary operation, for example, this might take the variables a, b, c, d, e and return the operation abc.

Remark 1.5. There are many natural ways to generalise the notion of Lawvere theory. Here are some examples.

1. We could use **FinSet** instead of the skeleton $\mathbb{F}$.
2. Many-sorted theories: writing $\mathcal{P}$ for the free finite-product category 2-monad on **Cat** and observing that $\mathcal{P}1 \simeq \mathbb{F}^{\mathrm{op}}$, we could instead use $\mathcal{P}A$ for non-terminal categories A to get Lawvere theories with sorts given by A.
3. Unsorted theories: we could just say that a Lawvere theory is *any* finite-product category C; in fact this can be regarded as a special case of many-sorted theories in which the sorts are given by the objects of C.
4. Enriched theories: we could use enriched categories, and get a notion of enriched Lawvere theory, and higher-dimensional Lawvere theory; see [11].
5. Φ-theories: we could use some other class Φ of limits than finite products, such as small products or finite limits; see [7].

While Lawvere theories enable us to study, say, the *theory* of groups as a mathematical object in its own right, models for Lawvere theories take us back to individual groups as mathematical objects.

Definition 1.6. A **model** for a Lawvere theory $\mathbb{L}$ in a finite-product category C is a finite-product preserving functor

$$\mathbb{L} \longrightarrow C.$$

A map of models is a natural transformation between them. These form a category $\mathbf{Mod}(\mathbb{L}, C)$.

Example 1.7. Let $\mathbb{L}$ be the Lawvere theory for monoids, and $C = \mathbf{Set}$. Consider a finite-product preserving functor

$$F \colon \mathbb{L} \longrightarrow C.$$

Writing $F(1) = A$, we must have $F(k) = A^k$. Then given any k-ary operation, that is, morphism $k \longrightarrow 1$ in $\mathbb{L}$, we get a function

$$A^k \longrightarrow A.$$

Functoriality and preservation of products ensures that this is precisely a monoid as expected. Putting $C = \mathbf{Top}$ gives an underlying *space* A with multiplication given by continuous maps, so we get topological monoids as expected.

We now discuss the correspondence between Lawvere theories and monads, which was hinted at in Example 1.4. This was originally analysed by Linton [10].

Proposition 1.8. *Given a monad T on* **Set** *we can construct a Lawvere theory $\mathbb{L}_T$ as the full subcategory of $\mathbf{Kl}T^{\mathrm{op}}$ whose objects are those of $\mathbb{F}$. Moreover*

$$\mathbf{Mod}(\mathbb{L}_T, \mathbf{Set}) \simeq \mathbf{Alg}T.$$

Remark 1.9. It is worth unravelling this a bit. Recall in Example 1.4 we saw that the morphisms $2 \longrightarrow 1$ in the Lawvere theory for monoids were given by all the elements of $T[2]$, where T is the free monoid monad and [2] is a 2-element set.

So we see that

$$\begin{array}{rcl} \mathbb{L}_T(2,1) & = & \mathbf{Set}(1, T[2]) \\ & = & \mathbf{Kl}T(1,2) \\ & = & \mathbf{Kl}T^{\mathrm{op}}(2,1). \end{array}$$

More generally a morphism $k \longrightarrow m$ is "m operations of arity k" ie

$$\begin{array}{rcl} \mathbb{L}_T(k,m) & = & \mathbf{Set}([m], T[k]) \\ & = & \mathbf{Kl}T([m],[k]) \\ & = & \mathbf{Kl}T^{\mathrm{op}}([k],[m]). \end{array}$$

Note that this has finite products because **Set** has coproducts. Now as we have only used finite sets, we cannot hope to have captured all the behaviour of a general monad on **Set**—only the finitary part. Recall that a finitary functor is one that preserves filtered colimits; on **Set** this amounts to being entirely determined by its action on finite sets as follows.

Proposition 1.10. *Let F be a functor* $\mathbf{Set} \longrightarrow \mathbf{Set}$. *Then F is finitary if and only if*

$$FX = \int^{[n]\in\mathbf{FinSet}} F[n] \times X^n.$$

This indicates how we can construct a monad from a Lawvere theory.

Proposition 1.11. *(Linton [10]) Given a Lawvere theory $\mathbb{L}$ we can construct a finitary monad on* **Set** *$T_{\mathbb{L}}$ by*

$$T_{\mathbb{L}}X = \int^{[n]\in\mathbf{FinSet}} \mathbb{L}(n,1) \times X^n.$$

This gives us a correspondence between Lawvere theories and finitary monads on **Set**.

Theorem 1.12. *The constructions $T \longmapsto \mathbb{L}_T$ and $\mathbb{L} \longmapsto T_{\mathbb{L}}$ extend to functors exhibiting* **Law** *as a full coreflective subcategory of* **Mnd***, the category of monads on* **Set***. Moreover, the essential image of the functor*

$$\mathbf{Law} \longrightarrow \mathbf{Mnd}$$

is given by the finitary monads, that is, the functor becomes an equivalence

$$\mathbf{Law} \xrightarrow{\simeq} \mathbf{Mnd}_f$$

where $\mathbf{Mnd}_f$ *denotes the full subcategory of finitary monads on* **Set***.*

This paper can be seen as providing several equivalent definitions of distributive law for Lawvere theory that extend the above correspondence.

2 Distributive laws for monads

In this work we will be thinking of distributive laws in two ways:

1. a way of combining algebraic theories to provide a composite theory, and
2. more generally: an abstract structure giving a way of composing monads to produce a composite monad inside any bicategory $\mathcal{B}$.

In this section we will simply recall the basic definitions. None of the material in this section is new. We first recall the classical theory of distributive laws.

Definition 2.1. (Beck [2])
Let S and T be monads on a category $\mathcal{C}$. A ***distributive law of*** *S* ***over*** *T consists of a natural transformation $\lambda\colon ST \Rightarrow TS$ such that the following diagrams commute.*

$$\begin{tikzcd} & T \arrow[dl, "\eta^S T"'] \arrow[dr, "T\eta^S"] & \\ ST \arrow[rr, "\lambda"'] & & TS \end{tikzcd} \qquad \begin{tikzcd} S^2T \arrow[r, "S\lambda"] \arrow[d, "\mu^S T"'] & STS \arrow[r, "\lambda S"] & TS^2 \arrow[d, "T\mu^S"] \\ ST \arrow[rr, "\lambda"'] & & TS \end{tikzcd} \tag{1}$$

$$\begin{tikzcd} & S \arrow[dl, "S\eta^T"'] \arrow[dr, "\eta^T S"] & \\ ST \arrow[rr, "\lambda"'] & & TS \end{tikzcd} \qquad \begin{tikzcd} ST^2 \arrow[r, "\lambda T"] \arrow[d, "S\mu^T"'] & TST \arrow[r, "T\lambda"] & T^2S \arrow[d, "\mu^T S"] \\ ST \arrow[rr, "\lambda"'] & & TS \end{tikzcd} \tag{2}$$

The main theorem about distributive laws tells us about new monads that arise canonically as a result of the distributive law. In this work we will mostly be interested in the composite monad.

Theorem 2.2 (Beck, [2])**.** *The following are equivalent:*

- *A distributive law of S over T.*

- *A lifting of the monad T to a monad T' on S-*Alg.
- *An extension of the monad S to a monad $\tilde{S}$ on* $\mathrm{Kl}(T)$.

It follows that TS canonically acquires the structure of a monad, whose category of algebras coincides with that of the lifted monad T', and whose Kleisli category coincides with that of $\tilde{S}$.

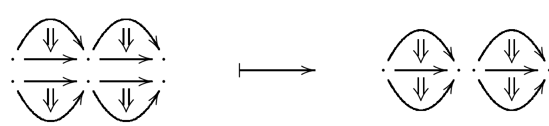
Example 2.3. (Rings)
$\mathcal{C} = \mathbf{Set}$
$S =$ free monoid monad
$T =$ free abelian group monad
$\lambda =$ the usual distributive law for multiplication and addition e.g.

$$(a+b)(c+d) \mapsto ac + bc + ad + bd.$$

Then the composite monad TS is the free ring monad.

Example 2.4. (2-categories)
$\mathcal{C} = \mathbf{2\text{-}GSet}$, the category of 2-globular sets.
$S =$ monad for vertical composition of 2-cells (1- and 0-cells are unchanged)
$T =$ monad for horizontal composition of 2-cells and 1-cells (0-cells are unchanged)
λ is given by the interchange law e.g.

$$ST \longrightarrow TS$$



The main theorem of [3] generalises the notion of distributive law to the case when we have more than two monads interacting with each other, as follows.

Theorem 2.5. *Fix $n \geq 3$. Let $T_1, \ldots, T_n$ be monads on a category $\mathcal{C}$, equipped with*

- *for all $i > j$ a distributive law $\lambda_{ij} : T_iT_j \Rightarrow T_jT_i$, satisfying*
- *for all $i > j > k$ the "Yang-Baxter" equation given by the commutativity of the following diagram*

$$\begin{tikzcd}
& T_jT_iT_k \arrow[r, "T_j\lambda_{ik}"] & T_jT_kT_i \arrow[dr, "\lambda_{jk}T_i"] & \\
T_iT_jT_k \arrow[ur, "\lambda_{ij}T_k"] \arrow[dr, "T_i\lambda_{jk}"'] & & & T_kT_jT_i \\
& T_iT_kT_j \arrow[r, "\lambda_{ik}T_j"'] & T_kT_iT_j \arrow[ur, "T_k\lambda_{ij}"'] &
\end{tikzcd} \tag{3}$$

Then for all $1 \leq i < n$ we have canonical monads

$$T_1 T_2 \cdots T_i \quad \textit{and} \quad T_{i+1} T_{i+2} \cdots T_n$$

together with a distributive law of $T_{i+1}T_{i+2}\cdots T_n$ *over* $T_1T_2\cdots T_i$ *i.e.*

$$(T_{i+1}T_{i+2}\cdots T_n)(T_1T_2\cdots T_i) \Rightarrow (T_1T_2\cdots T_i)(T_{i+1}T_{i+2}\cdots T_n)$$

given by the obvious composites of the λ_{ij}. *Moreover, all the induced monad structures on* $T_1T_2\cdots T_n$ *are the same.*

Definition 2.6. *A* ***distributive series of*** n ***monads*** *is a system of monads and distributive laws as in Theorem 2.5.*

Example 2.7. (Rings)
Rings can be constructed from the following distributive series of 3 monads on **Set**.

$$\begin{array}{rcl} A & = & \text{monad for associative non-unital binary multiplication } \times \\ B & = & \text{monad for pointed sets i.e. } X \mapsto X \coprod \{1\} \\ C & = & \text{free additive abelian group monad} \end{array}$$

Example 2.8. (Strict n-categories)
Strict n-categories can be constructed from a distributive series of n monads on n-globular sets, as a generalisation of the 2-category case. Here there is a monad T_i for each $0 \leq i \leq n-1$ giving "composition along bounding n-cells.

In his classic paper *The formal theory of monads* [13] Street defines for any 2-category $\mathcal{B}$ a 2-category $\mathbf{Mnd}(\mathcal{B})$ of monads in $\mathcal{B}$. Then distributive laws arise as monads in $\mathbf{Mnd}(\mathcal{B})$. While we will not use that particular, and appealing, fact, we will certainly be looking at monads and distributive laws inside various 2-categories and in fact bicategories, which can be done by invoking appropriate coherence conditions and results.

3 Monads in profunctors

In this section we give the most straightforward but perhaps least intuitive definition of distributive laws for Lawvere theories. We use the bicategory **Prof** of profunctors and simply observe that all Lawvere theories are monads on $\mathbb{F}^{\text{op}}$ in **Prof** (though not all monads on $\mathbb{F}^{\text{op}}$ are Lawvere theories). We can thus simply look at distributive laws between these monads. It is not immediately obvious why this should be the right definition and we will defer this justification to the last section.

First we set our notational conventions.

Definition 3.1. We write **Prof** for the bicategory given as follows.

- 0-cells are small categories,
- a 1-cell $\mathbb{C} \overset{F}{\longrightarrow\!\!\!\!\!\!\!\!+\;\;\;} \mathbb{D}$ is a functor $\mathbb{D}^{\text{op}} \times \mathbb{C} \longrightarrow \mathbf{Set}$,
- 2-cells are natural transformations.

Composition of profunctors $\mathbb{C} \xrightarrow{F} \mathbb{D} \xrightarrow{G} \mathbb{E}$ is by the usual coend formula

$$(G \circ F)(e,c) = \int^{d \in \mathbb{D}} G(e,d) \times F(d,c)$$

and is only weakly associative and unital.

Profunctors turn out to be the same as bimodules internal to the bicategory of spans. This fact will be useful to us both technically and conceptually in Section 4.

Definition 3.2. We write **Span** for the bicategory of spans given as follows.

- 0-cells are sets,
- a 1-cell $C \xrightarrow{X} D$ is a span

$$\begin{tikzcd} & X \arrow[dl, "s"'] \arrow[dr, "t"] & \\ C & & D \end{tikzcd}$$

- 2-cells are morphisms of spans.

Composition of 1-cells is by pullback, so given $C \xrightarrow{X} D \xrightarrow{Y} E$ we have

$$\begin{tikzcd} & & Y \circ X \arrow[dl] \arrow[dr] & & \\ & X \arrow[dl] \arrow[dr] & & Y \arrow[dl] \arrow[dr] & \\ C & & D & & E \end{tikzcd}$$

Definition 3.3. Given any bicategory K, we write **Mod**(K) for the bicategory of bimodules in K, given as follows.

- 0-cells are the monads in K,
- a 1-cell $X \xrightarrow{A} Y$ is a (Y,X)-bimodule (note direction).

  That is, if X is a monad $x \xrightarrow{X} x$ in K and Y is a monad $y \xrightarrow{Y} y$ in K then A is a 1-cell $x \xrightarrow{A} y$ in K equipped with 2-cell actions

$$\begin{tikzcd} x \arrow[r, "X"] \arrow[rr, bend right=50, "A"'] & x \arrow[r, "A"] \arrow[rr, bend left=50, "A"] \arrow[d, phantom, "\Downarrow\rho"] & y \arrow[r, "Y"] \arrow[u, phantom, "\Uparrow\lambda"] & y \end{tikzcd}$$

  satisfying the usual bimodule axioms: λ is compatible with the structure of X, ρ with the structure of Y and λ and ρ with each other.

- 2-cells are bimodule maps.

Composition of 1-cells is by coequaliser: given

$$X \xrightarrow{A}\!\!\!\!\!\!\circ\;\; Y \xrightarrow{B}\!\!\!\!\!\!\circ\;\; Z$$

given by 1-cells

$$x \xrightarrow{X} x \xrightarrow{A} y \xrightarrow{Y} y \xrightarrow{B} z \xrightarrow{Z} z$$

we take the coequaliser

$$\begin{tikzcd} B \circ Y \circ A \arrow[r, shift left, "\rho \circ A"] \arrow[r, shift right, "B \circ \lambda"'] & B \circ A \arrow[r] & B \otimes_Y A \end{tikzcd}$$

$B \otimes_y A$ is then the composite (Z, X)-bimodule required.

We now combine these two constructions and show that this gives another way of thinking of profunctors, with some care over dualities.

Example 3.4. The bicategory **Mod**(**Span**) is given as follows.

- 0-cells are monads in **Span** that is, small categories.
- Given categories X, Y with underlying spans

$$\begin{tikzcd} & X_1 \arrow[dl, "s"'] \arrow[dr, "t"] & \\ X_0 & & X_0 \end{tikzcd} \qquad \begin{tikzcd} & Y_1 \arrow[dl, "s"'] \arrow[dr, "t"] & \\ Y_0 & & Y_0 \end{tikzcd}$$

  a 1-cell $X \xrightarrow{A}\!\!\!\!\!\!\bullet\;\; Y$ has underlying span of the form

$$\begin{tikzcd} & A_1 \arrow[dl, "s"'] \arrow[dr, "t"] & \\ X_0 & & Y_0 \end{tikzcd}$$

  The elements of A can be thought of as arrows with source in X and target in Y. The left Y-action is a map of spans

$$\begin{tikzcd}[column sep=small] & & \cdot \arrow[dl] \arrow[dr] \arrow[dd, phantom, "\lrcorner" rotate=-45, very near start] & & \\ & A_1 \arrow[dl, "s"'] \arrow[dr, "t"] & & Y_1 \arrow[dl, "s"'] \arrow[dr, "t"] & \\ X_0 & & Y_0 & & Y_0 \end{tikzcd} \longrightarrow \begin{tikzcd} & A_1 \arrow[dl, "s"'] \arrow[dr, "t"] & \\ X_0 & & Y_0 \end{tikzcd}$$

  giving us a way of post-composing arrows in A with those of Y; the module axioms tell us that this respects composition in Y. Similarly for the left X-action. The left-right compatibility then gives us associativity for composing three arrows

$$\xrightarrow{\in X} \xrightarrow{\in A} \xrightarrow{\in Y}.$$

Proposition 3.5. There is a biequivalence of bicategories

$$\mathbf{Prof}^{\mathrm{op}} \simeq \mathbf{Mod}(\mathbf{Span}).$$

Proof. (Sketch.)

First we construct a functor

$$\mathbf{Prof}^{\mathrm{op}} \longrightarrow \mathbf{Mod}(\mathbf{Span}).$$

On 0-cells the functor is the identity.

For the action on 1-cells, we start with a profunctor $Y \overset{F}{\relbar\joinrel\mapstochar\joinrel\rightarrow} X$, that is, a functor $X^{\mathrm{op}} \times Y \xrightarrow{F} \mathbf{Set}$, and construct a bimodule $X \xrightarrow{\ \circ\ } Y$, that is, a (Y, X)-bimodule, as follows. First we need an underlying span $X \xrightarrow{\ \bullet\ } Y$, which is given as follows:

$$\begin{tikzcd}
 & \coprod\limits_{x,y} F(x,y) \arrow[dl, "s"'] \arrow[dr, "t"] & \\
X_0 & & Y_0
\end{tikzcd}$$

The left Y- and right X-actions are given by the actions of F on morphisms. For Y we need a map of spans

$$\begin{tikzcd}
 & & \cdot \arrow[dl] \arrow[dr] \arrow[d, phantom, "\lrcorner" rotate=-45] & & & & & \\
 & \coprod\limits_{x,y} F(x,y) \arrow[dl, "s"'] \arrow[dr, "t"] & & Y_1 \arrow[dl, "s"'] \arrow[dr, "t"] & & & \coprod\limits_{x,y} F(x,y) \arrow[dl, "s"'] \arrow[dr, "t"] & \\
X_0 & & Y_0 & & Y_0 \arrow[r, Rightarrow, phantom, "\longrightarrow"] & X_0 & & Y_0 .
\end{tikzcd}$$

An element in the pullback is a pair $(\alpha \in F(x,y), f \in Y_1(y,y'))$. Now we have

$$Ff\colon F(x,y) \longrightarrow F(x,y')$$

so we define the action by

$$(\alpha, f) \longmapsto Ff(\alpha)$$

and similarly for X.

Now we construct a functor

$$\mathbf{Mod}(\mathbf{Span}) \longrightarrow \mathbf{Prof}^{\mathrm{op}}$$

which again is the identity on 0-cells. Given categories X, Y and a bimodule $X \xrightarrow[\circ]{A} Y$, that is, a (Y, X)-bimodule with underlying span

$$\begin{tikzcd}
 & A_1 \arrow[dl, "s"'] \arrow[dr, "t"] & \\
X_0 & & Y_0,
\end{tikzcd}$$

say, we construct a profunctor $Y \relbar\joinrel\mapstochar\joinrel\rightarrow X$, that is, a functor

$$X^{\mathrm{op}} \times Y \xrightarrow{F} \mathbf{Set},$$

by $F(x,y) = A(x,y)$, that is, the pre-image in A_1 of the pair (x,y). Functoriality comes from the left and right actions. It is straightforward to check that this gives an equivalence. $\square$

Remarks 3.6.

1. Note that when we discuss factorisation systems in Section 4 it is useful to think in terms of spans, but for the comparison in Section 7 it is useful to think in terms of profunctors.

2. Note that given any bicategory $\mathcal{V}$ we can take $\mathbf{Mod}(\mathbf{Span}(\mathcal{V}))^{\mathrm{op}}$ as the *definition* of "internal profunctors in $\mathcal{V}$".

We are going to show that Lawvere theories arise as certain monads in **Prof**. In fact the monads in **Prof** are identity-on-objects functors. This is a special case of the following standard result.

Theorem 3.7. *Let K be a bicategory, x a 0-cell, and X a monad on x. Then there is an equivalence of categories*

$$\mathbf{Mon}\big((\mathbf{Mod}K)(X,X)\big) \simeq X/\mathbf{Mon}\big(K(x,x)\big).$$

Note that here we write $\mathbf{Mon}\mathcal{V}$ for the category of monoids in a monoidal category $\mathcal{V}$, and $\mathcal{B}(b,b)$ for the monoidal category of 1-cells $b \longrightarrow b$ in $\mathcal{B}$. Thus on the left hand side we

1. form the bicategory of bimodules in K,

2. take the monoidal category of 1-cells $X \longrightarrow\!\!\!\!\!\!\!\circ\;\; X$ in this bicategory, and

3. take the category of monoids in this monoidal category.

For the right hand side we

1. take the monoidal category of 1-cells $x \longrightarrow x$ in K,

2. take the category of monoids in this monoidal category, and

3. slice this category under X.

Corollary 3.8. *A monad in* $\mathbf{Prof}^{\mathrm{op}} \simeq \mathbf{Mod}(\mathbf{Span})$ *on X consists of a category A and an identity-on-objects functor $X \longrightarrow A$.*

Proof.
Put $K = \mathbf{Span}$, so X is a monad in K on the 0-cell $x = \mathrm{ob}\,X$. Then apply Theorem 3.7. Now a monad in $\mathbf{Mod}(\mathbf{Span})$ on X is a monoid in $\mathbf{Mod}(\mathbf{Span})(X,X)$ by definition, so by the theorem it is an object of $X/\mathbf{Mon}(\mathbf{Span})(x,x)$. Now

- a monoid in $\mathbf{Span}(x,x)$ is a category with the same objects as X, and

- a morphism of monoids in $\mathbf{Span}(x,x)$ is an identity-on-objects functor.

So $X/\mathbf{Mon}(K(x,x))$ has as objects identity-on-objects functors $X \longrightarrow A$. $\square$

Remark 3.9. It is illuminating to sketch a direct proof of this result. A monad $X \longrightarrow\!\!\!\!\!\!\!\!\!+\;\; X$ in $\mathbf{Mod}(\mathbf{Span})$ is an (X, X)-bimodule that is also a monad. That is, it has a left and right X-action but also a unit and multiplication of its own. Note that X is itself a monad in **Span**, with underlying span

$$\begin{tikzcd} & X_1 \arrow[dl] \arrow[dr] & \\ X_0 & & X_0 \end{tikzcd}$$

say. So for the monad $X \longrightarrow\!\!\!\!\!\!\!\!\!+\;\; X$ we have a span on the same objects as X, say

$$\begin{tikzcd} & A_1 \arrow[dl] \arrow[dr] & \\ X_0 & & X_0. \end{tikzcd}$$

Essentially

- the monad structure makes this into a category A, say,
- the left/right X-actions tell us how to map X_1 to A_1,
- the way composition of bimodules works ensures that the composition of A is compatible with that of X, that is, that we have a *functor* $X \longrightarrow A$.

Corollary 3.10. *Every Lawvere theory* $\mathbb{F}^{\mathrm{op}} \xrightarrow{\alpha_A} A$ *is a monad on* $\mathbb{F}^{\mathrm{op}}$ *in* $\mathbf{Prof}^{\mathrm{op}}$.

Of course, not every monad on $\mathbb{F}^{\mathrm{op}}$ in $\mathbf{Prof}^{\mathrm{op}}$ is a Lawvere theory—the category A must have finite products and the functor α_A must preserve them. However, given two Lawvere theories expressed in this way, we can define distributive laws between them.

Remark 3.11. At this point it might seem that we should have started with the opposite (dual) definition of **Prof**, which is also standard. However, in Section 6 we cannot use that version.

Definition 3.12. ("PROF") Given Lawvere theories A and B, a **distributive law** of A over B is a distributive law of A over B expressed as monads in $\mathbf{Prof}^{\mathrm{op}}$. Iterated distributive laws are defined likewise, as in Theorem 2.5.

Proposition 3.13. *The resulting composite monad BA is also a Lawvere theory.*

Note that the issue here is finite products—*a priori* our distributive law makes BA into a monad on $\mathbb{F}^{\mathrm{op}}$ in **Prof**, that is, an identity-on-objects functor $\mathbb{F}^{\mathrm{op}} \longrightarrow BA$; we still need to prove that BA has finite products and the functor preserves them. We defer this proof, and further justification of the definition, until Section 7 (Corollary 7.6), as the comparison proceeds via the definitions that we will introduce in subsequent sections.

In the next section we give a more explicit characterisation of such a distributive law, using the language of factorisation systems.

4 Factorisation systems

We will use a notion of factorisation system a little more general than that given by Rosebrugh and Wood in [12]. The stages of generalisation can be seen as follows:

1. Strict factorisation systems on a category C.
2. Factorisation systems over I where I is a subgroupoid of C; orthogonal factorisation systems are a special case.
3. Factorisation systems over J where J is a subcategory of C.

We include some basic definitions here as the terminology in the literature is not entirely uniform.

Definition 4.1. A **strict factorisation system** on a category C is a pair (L, R) of subcategories of C, with the same objects as C (lluf), such that every morphism of C can be factorised *uniquely* as a composite

$$\xrightarrow{l}\ \xrightarrow{r}$$

with $l \in L$ and $r \in R$.

Remarks 4.2.

1. The uniqueness implies that the intersection of L and R must contain only the identities.
2. It follows that $L \perp R$. (That is, every map in L has the *unique* left lifting property against every map in R, and every map in R has the *unique* right lifting property against every map in L; this means that lifts exist and are unique.)

Definition 4.3. An **orthogonal factorisation system** or simply **factorisation system** on a category C is a pair (L, R) of lluf subcategories of C, such that every morphism of C can be factorised as a composite

$$\xrightarrow{l}\ \xrightarrow{r}$$

with $l \in L$ and $r \in R$, uniquely up to unique isomorphism.

Remarks 4.4.

1. We do not actually need to stipulate that L and R are subcategories, as this follows from the rest of the structure.
2. It follows that $L \cap R$ must contain all isomorphisms. Thus if C contains non-trivial isomorphisms, a strict factorisation system on it is not an orthogonal factorisation system.
3. It follows that $L \perp R$ and in fact $L = {}^{\perp}R = {}^{\pitchfork}R$ and $R = L^{\perp} = L^{\pitchfork}$. Here we write $L^{\pitchfork}$ for the collection of maps with the right lifting property against all those in L, and $L^{\perp}$ for the collection of maps with the *unique* right lifting property against all those in L. Similarly for $L = {}^{\perp}R$ and ${}^{\pitchfork}R$ for left liftings.

Examples 4.5.

1. The pair ({epi}, {mono}) is an orthogonal factorisation system on **Set**.
2. The pair ({bijective-on-objects}, {full and faithful}) is an orthogonal factorisation system on **Cat**.
3. The pair ({bijective-on-objects and full}, {faithful}) is another orthogonal factorisation system on **Cat**.

There are not many naturally-arising strict factorisation systems, but the following characterisation by Rosebrugh and Wood [12] makes them of abstract interest.

Theorem 4.6. *Strict factorisation systems are precisely distributive laws in* **Span**. *That is, given a (small) category C, a strict factorisation system (A, B) on it is precisely a pair of monads A and B in* **Span** *together with a distributive law of A over B such that the composite monad BA is the category C.*

Another way of putting this is that a strict factorisation system on a category C is a decomposition of C as a monad in **Span** into a composite BA via a distributive law.

Remark 4.7. It is worth unravelling this a bit. The composite BA is a pullback. Writing underlying spans as

$$\begin{tikzcd} & A_1 \arrow[dl] \arrow[dr] & \\ X & & X \end{tikzcd}$$

and

$$\begin{tikzcd} & B_1 \arrow[dl] \arrow[dr] & \\ X & & X \end{tikzcd}$$

the composite BA is the pullback

$$\begin{tikzcd} & & \cdot \arrow[dl] \arrow[dr] \arrow[dd, phantom, "\lrcorner", very near start, rotate=-45] & & \\ & A_1 \arrow[dl] \arrow[dr] & & B_1 \arrow[dl] \arrow[dr] & \\ X & & X & & X \end{tikzcd}$$

and is not *a priori* a category. It consists of pairs of composable morphisms

$$\xrightarrow{\in A} \xrightarrow{\in B}.$$

The distributive law $AB \longrightarrow BA$ tells us how to re-express a composite

$$\xrightarrow{\in B} \xrightarrow{\in A}$$

as one in the "canonical form"

$$\xrightarrow{\in A} \xrightarrow{\in B}.$$

This makes BA into a category as we can now compose its morphisms: a composable pair in BA will be a composable quadruple

$$\xrightarrow{\in A}\xrightarrow{\in B}\xrightarrow{\in A}\xrightarrow{\in B}.$$

Then using the distributive law we can re-express the middle pair to get a string

$$\xrightarrow{\in A}\xrightarrow{\in A}\xrightarrow{\in B}\xrightarrow{\in B}$$

and we can then compose in A and in B separately to get a morphism in BA.

Note that morphisms in BA are *uniquely* expressible in the form

$$\xrightarrow{\in A}\xrightarrow{\in B}$$

by construction, as these are precisely the morphisms in the pullback.

Example 4.8. (Non-example)

It is instructive to note that this is not the notion we want for distributive laws of Lawvere theories. Let

$$\alpha\colon \mathbb{F}^{\mathrm{op}} \longrightarrow A$$

be the Lawvere theory for (multiplicative) monoids and

$$\beta\colon \mathbb{F}^{\mathrm{op}} \longrightarrow B$$

be the Lawvere theory for (additive) groups. Thus $X = \mathrm{ob}\,\mathbb{F}$. We will now see that BA does not give us the composite theory we want, namely, the theory of rings.

Consider the 3-ary operation $ab + c$ in the theory of rings. This certainly can be expressed as a composite

$$\xrightarrow{\in A}\xrightarrow{\in B}$$

via

$$3 \xrightarrow{\{ab,c\}} 2 \xrightarrow{x+y} 1.$$

However, this factorisation is not unique; for example we could also have

$$3 \xrightarrow{\{ab,c,abc\}} 2 \xrightarrow{x+y} 1.$$

where the first operation adds in a spurious operation and the second one forgets it. Now the two are related via a projection in $\mathbb{F}^{\mathrm{op}}$ making the following diagram commute, in the sense that the left-hand triangle commutes in A and the right-hand triangle commutes in B.

$$\begin{tikzcd}
 & 3 \arrow[dd, dashed, "{p_1, p_2}"'] \arrow[dr, "x+y"] & \\
3 \arrow[ur, "{ab, c, abc}"] \arrow[dr, "{ab, c}"'] & & 1 \\
 & 2 \arrow[ur, "x+y"'] &
\end{tikzcd}$$

The lesson is that we only want factorisations to be unique up to morphisms in $\mathbb{F}^{\mathrm{op}}$ somehow—in fact they are only unique up to zigzags in $\mathbb{F}^{\mathrm{op}}$. For example

the following two factorisations of the operation $a^2 + a^2$ cannot be related by a single morphism in $\mathbb{F}^{\mathrm{op}}$:

$$\begin{tikzcd}
& 3 \arrow[dr, "x+y"] \arrow[dd, dashed, "?"'] & \\
1 \arrow[ur, "{a^2, a^2, a}"] \arrow[dr, "a^2"'] & & 1 \\
& 1 \arrow[ur, "x+x"'] &
\end{tikzcd}$$

We will now show that there is no single morphism in $\mathbb{F}^{\mathrm{op}}$ in either direction ($3 \longrightarrow 1$ or $1 \longrightarrow 3$) that makes the diagram commute.

- For morphisms $3 \longrightarrow 1$, the only such morphisms are the three projections. These will clearly not make the resulting right-hand triangle commute.
- For morphisms $1 \longrightarrow 3$, the only such map is the diagonal $\{x, x, x\}$. This will not make the resulting left-hand triangle commute.

So in fact we need a zigzag:

$$\begin{tikzcd}
& 3 \arrow[d, dashed, "{p_1, p_2}"] \arrow[dr, "x+y"] & \\
1 \arrow[ur, "{a^2, a^2, a}"] \arrow[r, "{a^2, a^2}"] \arrow[dr, "a^2"'] & 2 \arrow[r, "x+y"] & 1 \\
& 1 \arrow[u, dashed, "\Delta"'] \arrow[ur, "x+x"'] &
\end{tikzcd}$$

where Δ denotes the diagonal.

Remark 4.9. Here is a useful way of thinking about this example that points us in the direction we need. The idea is that our original pullback BA

$$\begin{tikzcd}
& & \cdot \arrow[dl] \arrow[dr] & & \\
& A_1 \arrow[dl] \arrow[dr] & & B_1 \arrow[dl] \arrow[dr] & \\
X & & X & & X
\end{tikzcd}$$

ignored the fact that $\mathbb{F}^{\mathrm{op}}$ is in both A and B. So in fact we want a coequaliser

$$\begin{tikzcd}
B \circ \mathbb{F}^{\mathrm{op}} \circ A \arrow[r, shift left] \arrow[r, shift right] & B \circ A \arrow[r] & B \otimes_{\mathbb{F}^{\mathrm{op}}} A
\end{tikzcd}$$

where the parallel maps are derived from

$$\mathbb{F}^{\mathrm{op}} \xrightarrow{\alpha} A, \text{ and}$$

$$\mathbb{F}^{\mathrm{op}} \xrightarrow{\beta} B$$

respectively. To form this coequaliser we put an equivalence relation on the morphisms of BA; this is encapsulated in the following definition.

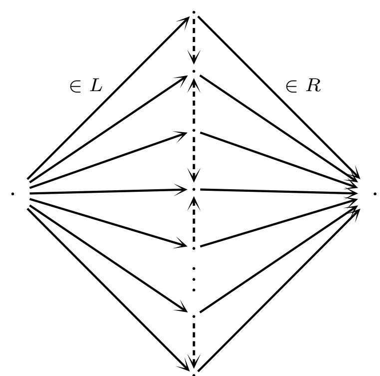
Definition 4.10. Let C be a category, J a subcategory with the same objects as C (lluf). A **factorisation system over** J on C consists of

- a lluf subcategory L of C containing J, and
- a lluf subcategory R of C containing J

such that every morphism in C can be expressed as

$$\xrightarrow{\in L}\xrightarrow{\in R}$$

uniquely up to zigzags in J as shown in the following diagram, where the morphisms on the left hand half of the diagram are all in L, those on the right are all in R, and the vertical dotted morphisms are in J. The triangles on the left commute in L and those on the right commute in R.



Examples 4.11.

1. If J is a groupoid, we get a factorisation system over J as in [12]. (In fact they stop just short of making this definition although they have all the machinery in place to make it—they make the following construction instead.)
2. If J is the groupoid of all isomorphisms in C, we get an orthogonal factorisation system in the usual sense.
3. If J is all identities we get a strict factorisation system.
4. Weak factorisation systems are not in general an example, for in a weak factorisation system factorisations are unique up to diagonal fillers, or "solutions" of certain lifting problems, but these diagonal fillers are not necessarily in L or R; to be a factorisation system over J these fillers would need to be in J and hence in both L and R.

Definition 4.12. ("FS") Let A, B and C be Lawvere theories. Then we say C is a **composite** of A and B if (A, B) forms a factorisation system over $\mathbb{F}^{\mathrm{op}}$ on C. In this case we say we have a **distributive law** of A over B.

Proposition 4.13. *Given any category C with a factorisation system over $\mathbb{F}^{\mathrm{op}}$ given by (A, B), if A and B are Lawvere theories then C is also a Lawvere theory.*

As before (for the definition in $\mathbf{Prof}^{\mathrm{op}}$), we need to check the necessary facts about finite products. Again we defer this proof until later (Corollary 7.6).

Remark 4.14. Note that the natural way of stating this definition involved starting with a category C and "decomposing it" via a factorisation system over $\mathbb{F}^{\mathrm{op}}$, rather than starting with Lawvere theories A and B and "combining them" as in other definitions. This different viewpoint could shed light on the question of when an algebraic theory can can be expressed as a composite of simpler ones, as opposed to when it is "irreducible".

In any case the formulation as a coequaliser gives us a good abstract formalism. Effectively we have taken the monoidal category $\mathbf{Span}(\mathbb{F}^{\mathrm{op}}, \mathbb{F}^{\mathrm{op}})$, put a new tensor product $\otimes_{\mathbb{F}^{\mathrm{op}}}$ on it, and taken distributive laws with respect to this. This is more elegantly described using bimodules.

Proposition 4.15. *A Lawvere theory*

$$\mathbb{F}^{\mathrm{op}} \xrightarrow{\alpha} A$$

is an $(\mathbb{F}^{\mathrm{op}}, \mathbb{F}^{\mathrm{op}})$*-bimodule in* $\mathbf{Span}$*.*

Then $\otimes_{\mathbb{F}^{\mathrm{op}}}$ described above is just bimodule composition. Thus the above definition of distributive law amounts to regarding A and B as monads in $\mathbf{Mod}(\mathbf{Span})$ and taking distributive laws between them. But we know $\mathbf{Mod}(\mathbf{Span}) \simeq \mathbf{Prof}^{\mathrm{op}}$, so we have proved the following theorem.

Theorem 4.16. *Distributive laws as in Definition 4.12* "FS" *are equivalent to distributive laws as in Definition 5.5* "PROF"*.*

We will state this more precisely later in terms of comparison functors, but the idea is that Definition 4.12 "FS" can be taken as an explicit characterisation of Definition 5.5 "PROF".

Remarks 4.17.

1. This definition generalises the definition of "distributive law with respect to J" given in [12] although there it is expressed quite differently. J is required to be a groupoid in order to yield an equivalence relation on the morphisms of BA. Effectively, this is to get unique factorisations up to plain morphisms in J rather than zigzags (see [12, Section 5.4]). In fact the authors do not actually mention factorisation systems over general groupoids—their aim is to give a bicategory in which orthogonal factorisation systems are distributive laws, so once they have this general notion of distributive law in place, they set J to be the groupoid of all isomorphisms for the purposes of the factorisation system.

2. Lack discusses a version of this in [8, Sections 4.2, 4.3]. He is mostly concerned with PROPs, so only mentions this in passing, and again only in the case where J is a groupoid. However, his subsequent sections study distributive laws in $\mathbf{Prof}(\mathbf{Mon})$, which is also the subject of our next section.

5 Monads in monoidal profunctors

In this section we give an approach that deals a little more explicitly with the finite products.

Definition 5.1. Let $\mathcal{E}$ be a category with pullbacks and coequalisers that commute.

- Write $\mathbf{Span}\,\mathcal{E}$ for the bicategory of spans in $\mathcal{E}$.
- Write $\mathbf{Prof}\,\mathcal{E}$ for the bicategory $\mathbf{Mod}(\mathbf{Span}\,\mathcal{E})^{\mathrm{op}}$. Thus 0-cells are monads in $\mathbf{Span}\,\mathcal{E}$, that is, categories internal to $\mathcal{E}$.

Remarks 5.2.

1. We need pullbacks to define composition of spans, and we need the coequaliser condition to define composition of profunctors.
2. We need to take the dual here for the same reason as in the previous section.

The example we will be using in this section is the case $\mathcal{E} = \mathbf{Mon}$, the category of monoids and monoid homomorphisms. Note that a 0-cell in $\mathbf{ProfMon}$ is an internal category in $\mathbf{Mon}$, that is, a strict monoidal category.

Proposition 5.3. *A monad in* $\mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}$ *on a monoidal category* X *consists of a strict monoidal category* A *and an identity-on-objects strict monoidal functor* $X \longrightarrow A$.

Proof. Follows from Theorem 3.7. Put $K = \mathbf{Span}(\mathbf{Mon})$, and $x = \mathbf{ob}\,X$, so a monoid in $K(x, x)$ in this case is a strict monoidal category with the same objects as X. A morphism of such monoids is a strict monoidal identity-on-objects functor. $\square$

The following result is analogous to Corollary 3.10.

Corollary 5.4. *Every Lawvere theory is a monad in* $\mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}$ *on the 0-cell* $\mathbb{F}^{\mathrm{op}}$ *regarded as a monoidal category with respect to product.*

Note that, as in $\mathbf{Prof}^{\mathrm{op}}$, not every monad

$$\alpha : \mathbb{F}^{\mathrm{op}} \longrightarrow A$$

in $\mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}$ is a Lawvere theory. Given such a monad, A is strict monoidal, but its monoidal structure is not necessarily given by finite products. However, if it *is* given by finite products then α must preserve them (as it is strict monoidal). In this sense the framework is slightly better than working in plain $\mathbf{Prof}$; in the next section we will give an even "better" framework.

Definition 5.5. ("PROFMON") Given Lawvere theories A and B, a **distributive law** of A over B is a distributive law of A over B expressed as monads in $\mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}$. The iterated version is defined likewise, as in Theorem 2.5.

Proposition 5.6. *The resulting composite monad* $B \otimes_{\mathbb{F}^{\mathrm{op}}} A$ *is also a Lawvere theory.*

Remarks 5.7.

1. An immediate question is whether or not this gives the same thing as distributive laws in plain $\mathbf{Prof}$. The slightly surprising answer is that they are indeed the same, as when the monoidal structure is product, natural transformations are automatically monoidal. We will discuss this in Section 7.

2. This approach is closely related to Lack's approach to distributive laws for PROPs in [8]. For PROPs, instead of $\mathbb{F}$ we use $\mathbb{P}$, a skeleton of the the category of finite sets and *bijections*. The rest of the formalism is the same. This approach was mentioned in [1].

As before, we defer the proof that the composite is a Lawvere theory until Section 7, but it is instructive to compare the question to the analogous question in $\mathbf{Prof}^{\mathrm{op}}$. There, the issue was both whether the composite had finite products and whether the identity-on-objects functor preserved them. This time, we know the functor must preserve the monoidal structure of the composite, so we only need to check that this monoidal structure is given by finite products.

There are (at least) two ways to prove this. One is a direct hands-on method alluded to in [1]; a more abstract approach uses a free finite-product category 2-monad. This is the subject of the next section.

6 Monads in a Kleisli bicategory of profunctors

In this section we use a bicategory in which monads are *precisely* Lawvere theories. (This statement allows for types—for plain Lawvere theories we can restrict to the 0-cell 1 as we shall see.) We use the free finite-product category 2-monad as follows.

Proposition 6.1. ***(Hyland [5])*** *Write $\mathcal{P}$ for the free finite-product category 2-monad on* **Cat***. This extends to a 2-monad on* **Prof***.*

Proof.(Sketch)

Write S for the preseheaf functor, so $SC = [C^{\mathrm{op}},, \mathbf{Set}]$. Aside from size issues, this is a 2-monad but for—the 2-monad for colimit completion. Its Kleisli bicategory is equivalent to **Prof** since a profunctor $D^{\mathrm{op}} \times C \longrightarrow \mathbf{Set}$ is equivalently a functor $C \longrightarrow [D^{\mathrm{op}}, \mathbf{Set}] = SD$.

Now by Beck's theorem in 2-dimensions [4] an extension of $\mathcal{P}$ to the bicategory $\mathbf{Kl}S$ is equivalent to a lift of S to $\mathbf{Alg}\,\mathcal{P}$. But $\mathbf{Alg}\,\mathcal{P}$ is just the 2-category of finite-product categories.

To exhibit such a lift it suffices to show that if C is a P-algebra:

1. SC is a P-algebra,
2. the Yoneda embedding (ie unit for S) $C \longrightarrow SC$ is a P-algebra map, and
3. given a P-algebra map $f\colon C \xrightarrow{f} SC$, the left Kan extension along the Yoneda embedding

$$\begin{tikzcd} C \arrow[r, "\text{Yoneda}"] \arrow[d, "f"'] & SC \arrow[dl, dashed, "f^\dagger"] \\ SD & \end{tikzcd}$$

is a P-algebra map.

Here (1) and (2) are straightforward as presheaf categories have products and the Yoneda embedding preserves them. For (3) we use the coend formula for left Kan extensions:

$$f^{\dagger}(X) = \int^{c \in C} f(c) \times SC(H^c, X)$$

where H^c denotes the appropriate representable. The result is then true since coends commute with finite products. $\square$

By abuse of notation we will write the extended monad as $\mathcal{P}$; this should not cause ambiguity as we will never need to use the original monad on **Cat**.

Remarks 6.2. It is useful to take a moment to make some of the structure of $\mathcal{P}$ explicit; we will need this in the proof of Proposition 6.5.

1. First we make explicit the structure of $\mathcal{P}A$ where A is any category. Objects in $\mathcal{P}A$ are finite strings of objects in A. Since these are to be products, a morphism
$$(a_1, \ldots, a_n) \longrightarrow (b_1, \ldots b_m)$$
is given by
   - a function $\alpha\colon [m] \longrightarrow [n]$, and
   - for each $i \in [m]$ a morphism $a_{\alpha(i)} \longrightarrow b_i$ in A.

   In the proof of Proposition 6.5 we will need the morphisms of $\mathcal{P}^2 1$. An object in this category is a string of natural numbers. We see that in this case a morphism
$$(a_1, \ldots, a_n) \longrightarrow (b_1, \ldots b_m)$$
is given by
   - a function $\alpha\colon [m] \longrightarrow [n]$, and
   - for each $i \in [m]$ a function $[b_i] \longrightarrow [a_{\alpha(i)}]$.

2. Next we give the action of $\mathcal{P}$ on morphisms. Given a profunctor
$$F\colon A \longrightarrow\!\!\!\!\!\!\!+\;\; B \quad \text{i.e.} \quad B^{\text{op}} \times A \longrightarrow \mathbf{Set}$$
we need a profunctor
$$\mathcal{P}F\colon \mathcal{P}A \longrightarrow\!\!\!\!\!\!\!+\;\; \mathcal{P}B \quad \text{i.e.} \quad \mathcal{P}B^{\text{op}} \times \mathcal{P}A \longrightarrow \mathbf{Set}.$$
The profunctor $\mathcal{P}F$ is defined by
$$\begin{aligned}
\mathcal{P}F(b_1, \ldots, b_n; a_1, \ldots, a_n) &= \coprod_{\alpha \in \mathbf{Set}(m,n)} \prod_{j \in [m]} F(b_{\alpha_j}, a_j) \\
&= \prod_{j \in [m]} \coprod_{i \in [n]} F(b_i, a_j) \\
&= \prod_{j \in [m]} \left(\coprod_{i \in [n]} F(b_i, a_j) \right)^m
\end{aligned}$$

3. Next we give the monad structure. For multiplication we have

$$\mu \colon \mathcal{P}^2 1 \longrightarrow\!\!\!\!\!\!\!\!\;\;+\;\;\; \mathcal{P}1 \quad \text{i.e.} \quad \mathcal{P}1^{\,\mathrm{op}} \times \mathcal{P}^2 1 \longrightarrow \mathbf{Set}$$

given by

$$\mu(n; k_1, \ldots, k_m) = \mathcal{P}1(n, k_1 + \cdots + k_m).$$

For the unit we have

$$\eta \colon 1 \longrightarrow\!\!\!\!\!\!\!\!\;\;+\;\;\; \mathcal{P}1 \quad \text{i.e.} \quad \mathcal{P}1^{\,\mathrm{op}} \longrightarrow \mathbf{Set}$$

given by

$$\eta(k) = \mathcal{P}1(k, 1) = \mathbf{Set}(1, [k]).$$

Definition 6.3. Write $\mathbf{Prof}_{\mathcal{P}}$ for the Kleisli bicategory of $\mathcal{P}$ extended to $\mathbf{Prof}$.

Theorem 6.4. ***(Hyland)*** *Monads on 1 in* $\mathbf{Prof}_{\mathcal{P}}$ *are precisely Lawvere theories.*

Proof. (Sketch.)

A monad on 1 in $\mathbf{Prof}_{\mathcal{P}}^{\mathrm{op}}$ has an underlying profunctor $1 \longrightarrow\!\!\!\!\!\!\!\!\;\;+\;\;\; \mathcal{P}1$, that is, a functor

$$\mathcal{P}1^{\,\mathrm{op}} \times 1 \simeq \mathbf{FinSet} \longrightarrow \mathbf{Set}$$

The monad structure makes this into a finitary monad on $\mathbf{Set}$. $\square$

In fact we have a more precise result involving an equivalence of categories (Theorem 6.7). Before we prove that, the following proposition provides a functor that will evaluate a monad in $\mathbf{Prof}_{\mathcal{P}}^{\mathrm{op}}$ at the corresponding Lawvere theory expressed in $\mathbf{Prof}^{\mathrm{op}}$. Recall that the forgetful functor from the Kleisli category of any monad to its underlying category is given on morphisms by applying the monad and postcomposing with μ. The following proposition evaluates this for $\mathcal{P}$.

Proposition 6.5. *For any profunctor* $1 \xrightarrow{F} \mathcal{P}1$*, the composite*

$$\mathcal{P}1 \xrightarrow{\mathcal{P}F} \mathcal{P}^2 1 \xrightarrow{\mu} \mathcal{P}1$$

is given by

$$\begin{array}{ccc} \mathcal{P}1^{\,\mathrm{op}} \times \mathcal{P}1 & \longrightarrow & \mathbf{Set} \\ (j, n) & \longmapsto & \mathbf{Set}(n, Fj) \end{array}.$$

Proof. By definition this composite is

$$\begin{array}{ccl} (j, l) & \longmapsto & \displaystyle\int^{(k_1, \ldots, k_m) \in \mathcal{P}^2 1} \mu(j; k_1 + \cdots + k_m) \times \mathcal{P}F(k_1, \ldots, k_m; l) \\ & = & \displaystyle\int^{(k_1, \ldots, k_m) \in \mathcal{P}^2 1} \mathcal{P}1(j, k_1 + \cdots + k_m) \times \left(\coprod_{i \in [m]} F(k_i) \right)^{l} \end{array}$$

We aim to show that in computing this coend we only need to consider $m = 1$. We use the fact that in general in a coend cocone for $Q : \mathbb{I}^{\mathrm{op}} \times \mathbb{I} \longrightarrow \mathbf{Set}$

$$\begin{tikzcd}[row sep=small] & Q(U,U) \arrow[dr] & \\ Q(V,U) \arrow[ur, "Q(f,1)"] \arrow[dr, "Q(1,f)"'] & & \cdot \\ & Q(V,V) \arrow[ur] & \end{tikzcd}$$

for $f\colon U \longrightarrow V$ in $\mathbb{I}$, if $Q(1, f)$ is surjective we can ignore $Q(V, V)$ as no further information is contributed by it.

Choose $U, V \in \mathcal{P}^2 1$ as follows

$$\begin{array}{rcl} U & = & (k_1 + \cdots + k_m) = (k), \quad \text{say} \\ V & = & (k_1, \ldots, k_m). \end{array}$$

Note that U is a 1-ary string. We then define $f\colon U \longrightarrow V \in \mathcal{P}^2 1$ as follows. Recall that a morphism

$$(a_1, \ldots, a_n) \longrightarrow (b_1, \ldots, b_m)$$

in $\mathcal{P}^2 1$ consists of

- a map $\alpha\colon m \longrightarrow n$ in $\mathbf{Set}$, and
- for all $i \in [m]$, a map $\beta_i\colon b_i \longrightarrow a_{\alpha(i)}$ in $\mathbf{Set}$.

Here we have $n = 1$ so α is trivial, thus to define f we just need to give, for all $i \in [m]$ a map

$$\beta_i\colon k_i \longrightarrow k_1 + \cdots + k_m \in \mathbf{Set}$$

and we set these to be the canonical coproduct insertions.

Now note that

$$\begin{array}{rcl} Q(V,U) & = & \mathcal{P}1(j, k_1 + \cdots + km) \times \left(\displaystyle\coprod_{i \in [m]} F(k_i) \right)^l \\ & \cong & Q(V,V) \end{array}$$

and moreover the isomorphism is given by $Q(1, f)$. So we can disregard all vertices in the coend for which $m \neq 1$.

Thus the coend becomes

$$\int^{k \in \mathcal{P}1} \mathcal{P}1(j,k) \times \big(F(k)\big)^l \cong \mathbf{Set}(n, Fj)$$

as required. □

Remark 6.6. Note that this profunctor will be called $\bar{F}$ in Section 7 and it will give us the comparison between the profunctor approach and the monad approach; note that if F is a finitary monad, $\bar{F}$ is its associated Lawvere theory.

Write $\mathbf{CAT}_f$ for the (large) 2-category of locally small categories, finitary functors and natural transformations.

Theorem 6.7. *There is a monoidal equivalence of categories*

$$\mathbf{CAT}_f(\mathbf{Set}, \mathbf{Set}) \xrightarrow{\simeq} \mathbf{Prof}_{\mathcal{P}}^{\mathrm{op}}(1, 1).$$

Proof. Recall that a finitary functor $F \colon \mathbf{Set} \longrightarrow \mathbf{Set}$ is entirely determined by its restriction to **FinSet**, by the formula

$$FX = \int^{[n] \in \mathbf{FinSet}} F[n] \times X^n.$$

We define a functor

$$\mathbf{CAT}_f(\mathbf{Set}, \mathbf{Set}) \xrightarrow{\theta} \mathbf{Prof}_{\mathcal{P}}^{\mathrm{op}}(1, 1)$$

as follows. Given a finitary functor $F : \mathbf{Set} \longrightarrow \mathbf{Set}$ we restrict it as

$$\mathcal{P}1^{\mathrm{op}} \simeq \mathbf{FinSet} \xrightarrow{F} \mathbf{Set}$$

which can be regarded as a profunctor $1 \xrightarrow{\theta F} \mathcal{P}1$ as required. (Note that technicall we must pick a functor $\mathcal{P}1^{\mathrm{op}} \longrightarrow \mathbf{FinSet}$ giving the equivalence.)

On morphisms we also take the restriction of natural transformations to **FinSet**.

The interesting part is the monoidal structure, which is given by composition. Consider finitary functors

$$\mathbf{Set} \xrightarrow{F} \mathbf{Set} \xrightarrow{G} \mathbf{Set}.$$

Then the composite $\theta G \circ \theta F$ in $\mathbf{Prof}_{\mathcal{P}}$ is given by the profunctor composite

$$1 \xrightarrow{\theta F} \mathcal{P}1 \xrightarrow{\mathcal{P}(\theta G)} \mathcal{P}^2 1 \xrightarrow{\mu} \mathcal{P}1$$

which is some functor

$$\mathcal{P}1^{\mathrm{op}} \longrightarrow \mathbf{Set}.$$

Now, using the formula for μ and the action of $\mathcal{P}$ on morphisms as given in Remarks 6.2 we see that the composite is given by

$$\begin{array}{rcll} m & \longmapsto & \displaystyle\int^{j \in \mathcal{P}1} \mathbf{Set}(j, \theta G(m)) \times \theta F(j) & \text{by Proposition 6.5} \\ & = & \displaystyle\int^{j \in \mathcal{P}1} \mathbf{Set}(j, Gm) \times F(j) & \\ & = & FG(m) & \text{by standard density} \\ & = & \theta(FG)(m) & \end{array}$$

It remains to show that this functor is essentially surjective; full and faithfulness is clear. But a finitary functor F is determined uniquely up to isomorphism by its restriction to **FinSet**, so the result is true. □

Definition 6.8. "KLEISLI" *Given Lawvere theories A and B, a **distributive law of** A **over** B is a distributive law of A over B expressed as monads on 1 in $\mathbf{Prof}_{\mathcal{P}}^{\mathrm{op}}$. The composite monad BA is automatically a Lawvere theory, and is called the **composite Lawvere theory**. The iterated version is defined likewise, as in Theorem 2.5.*

Note that this is the only case in which it is immediate that the composite monad is a Lawvere theory; however the result for the other definitions will follow. First, we can immediately deduce from the preceding results that these distributive laws correspond precisely to distributive laws between finitary monads in **Set**.

Corollary 6.9. *Let S and T be finitary monads on* **Set** *with associated Lawvere theories*

$$\begin{array}{rcl} \theta(S) & = & \mathbb{L}_S \\ \theta(T) & = & \mathbb{L}_T \end{array}$$

expressed as monads on 1 in $\mathbf{Prof}_{\mathcal{P}}^{\mathrm{op}}$. Let

$$\lambda\colon ST \Longrightarrow TS$$

be a distributive law of S over T. Then

$$\theta(\lambda)\colon \theta(ST) \Longrightarrow \theta(TS)$$

gives a distributive law of $\mathbb{L}_S$ over $\mathbb{L}_T$ in $\mathbf{Prof}_{\mathcal{P}}^{\mathrm{op}}$ as

$$\begin{array}{rcl} \theta(ST) & \cong & \mathbb{L}_S\mathbb{L}_T \\ \theta(TS) & \cong & \mathbb{L}_T\mathbb{L}_S. \end{array}$$

Furthermore since $\mathbb{L}_{TS} = \theta(TS) \cong \mathbb{L}_T\mathbb{L}_S$ we see that the composite Lawvere theory is the Lawvere theory associated to the composite monad. Conversely since θ is an equivalence, every distributive law of Lawvere theories arises in this way.

Remark 6.10. In fact since distributive laws in K are the 0-cells of $\mathbf{Mnd}(\mathbf{Mnd}\,K)$ we could express this as a biequivalence between the "bicategories of distributive laws", and then iterate the construction to get a notion of iterated distributive law for Lawvere theory, as in Definition 2.5.

7 Comparison

We now have four definitions of distributive law for Lawvere theory in place:

1. PROF: Distributive laws in $\mathbf{Prof}_{\mathcal{P}}^{\mathrm{op}}$.
2. FS: Factorisation systems over $\mathbb{F}^{\mathrm{op}}$.
3. PROFMON: Distributive laws in $\mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}$.
4. KL: Distributive laws in $\mathbf{Prof}_{\mathcal{P}}^{\mathrm{op}}$.

So far we have shown that

- PROF and FS are equivalent (Theorem 4.16).
- KL is equivalent to the monad approach (Corollary 6.9).

In this section we will complete the programme of equivalences by showing that PROF is equivalent to both PROFMON and the monad approach. We will have the following diagram summing up our comparisons

$$\begin{tikzcd}
& \mathbf{Prof}^{\mathrm{op}}_{\mathcal{P}}(1,1) \arrow[d, "U^{\theta}\ \text{forgetful}"] \\
& \mathbf{Prof}^{\mathrm{op}}(\mathcal{P}1,\mathcal{P}1) \arrow[d, phantom, "\wr"] \\
\mathbf{CAT}_f(\mathbf{Set},\mathbf{Set}) \arrow[uur, "\theta\ \text{equivalence}"] \arrow[r, "\phi\ \text{f+f}"] \arrow[dr, "\psi\ \text{f+f}"'] & \mathbf{Prof}^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}},\mathbb{F}^{\mathrm{op}}) \\
& \mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}},\mathbb{F}^{\mathrm{op}}) \arrow[u, "U^{\psi}\ \text{forgetful}"']
\end{tikzcd}$$

We sum up our strategy (including results we have already proved) as follows.

1. We prove we have a functor
$$\phi\colon \mathbf{CAT}_f(\mathbf{Set},\mathbf{Set}) \longrightarrow \mathbf{Prof}^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}},\mathbb{F}^{\mathrm{op}})$$
sending finitary monads to their associated Lawvere theories, which is full and faithful. This will be done in this section.

2. This functor factors through $\mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}$.
$$\begin{tikzcd}
\mathbf{CAT}_f(\mathbf{Set},\mathbf{Set}) \arrow[r, "\phi"] \arrow[dr, "\psi"'] & \mathbf{Prof}^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}},\mathbb{F}^{\mathrm{op}}) \\
& \mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}},\mathbb{F}^{\mathrm{op}}) \arrow[u, dashed, "U^{\psi}"']
\end{tikzcd}$$
The factor ψ must therefore also be full, and is obviously faithful. Thus the forgetful functor U^{ψ} must also be full and faithful. This shows that PROF is equivalent to PROFMON.

3. We have already exhibited a monoidal equivalence
$$\theta\colon \mathbf{CAT}_f(\mathbf{Set},\mathbf{Set}) \xrightarrow{\simeq} \mathbf{Prof}^{\mathrm{op}}_{\mathcal{P}}(1,1)$$
showing that KL is equivalent to the monad approach (Theorem 6.7 and Corollary 6.9).

4. Proposition 6.5 shows that the canonical Kleisli forgetful functor U^{θ} makes the following triangle commute (up to isomorphism)
$$\begin{tikzcd}
& \mathbf{Prof}^{\mathrm{op}}_{\mathcal{P}}(1,1) \arrow[d, dashed] \\
\mathbf{CAT}_f(\mathbf{Set},\mathbf{Set}) \arrow[ur, "\theta"] \arrow[r, "\phi"] & \mathbf{Prof}^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}},\mathbb{F}^{\mathrm{op}})
\end{tikzcd}$$

As the forgetful functor is a 1-object restriction of a pseudo-functor, it must be monoidal. Thus the functor ϕ must be monoidal. This completes the proof that PROF is equivalent to the monad approach.

5. Since θ and ϕ are full and faithful, the forgetful functor must also be full and faithful, giving a direct comparison between the KLEISLI and PROF approaches.

This will complete the suite of equivalences; it remains to complete the parts we have not already proved.

First we define a functor

$$
\begin{array}{ccc}
\phi\colon \mathbf{CAT}_f(\mathbf{Set},\mathbf{Set}) & \longrightarrow & \mathbf{Prof}^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}},\mathbb{F}^{\mathrm{op}}) \\
F & \longmapsto & \bar{F}\colon \mathbb{F}\times\mathbb{F}^{\mathrm{op}} \longrightarrow \mathbf{Set} \\
 & & \bar{F}(n,m) = \mathbf{Set}(m,Fn) \\
\alpha : F \Longrightarrow G & \longmapsto & \begin{tikzcd} \mathbb{F}\times\mathbb{F}^{\mathrm{op}} \arrow[r, bend left, "\bar{F}"] \arrow[r, bend right, "\bar{G}"'] \arrow[r, phantom, "\Downarrow\bar{\alpha}"] & \mathbf{Set} \end{tikzcd} \\
\alpha_n\colon Fn \longrightarrow Gn & & \bar{\alpha}_{n,m}\colon \bar{F}(n,m) \longrightarrow \bar{G}(n,m) \\
 & & \mathbf{Set}(m,Fn) \xrightarrow{\alpha_n\circ\_} \mathbf{Set}(m,Gn)
\end{array}
$$

Later we will show that this is a monoidal functor, but now we concentrate on other properties.

Proposition 7.1. *The functor ϕ is clearly faithful (by Yoneda). It is also full.*

Note that this result is perhaps surprising. Even if we consider only the monads on each side, we find that monads on the left give a particular kind of monad on the right—those identity-on-objects functors

$$\alpha\colon \mathbb{F}^{\mathrm{op}} \longrightarrow A$$

where A has finite products and α preserves finite products. Now maps between monads on the left only give maps on the right

$$\tau\colon A \longrightarrow A'$$

that preserve finite products. Thus at first sight it might seem that the functor should not be full, as there should be maps τ on the right that do not preserve finite products. However this is not the case.

Proof. Suppose we have a natural transformation

$$\begin{tikzcd} \mathbb{F}\times\mathbb{F}^{\mathrm{op}} \arrow[r, bend left, "\bar{F}"] \arrow[r, bend right, "\bar{G}"'] \arrow[r, phantom, "\Downarrow\beta"] & \mathbf{Set}. \end{tikzcd}$$

We aim to show that β is in fact of the form $\bar{\alpha}$ for some α as above. Now, given any $n \in \mathbb{F}$ we have

$$\begin{array}{cccc} \beta_{n,1} : & \bar{F}(n,1) & \longrightarrow & \bar{G}(n,1) \\ \text{ie} & \mathbf{Set}(1, Fn) & \longrightarrow & \mathbf{Set}(1, Gn) \\ & \| & & \| \\ & Fn & & Gn \end{array}$$

Call this α_n.

We claim

1. These α_n are the components of a natural transformation $\alpha\colon F \Longrightarrow G$, and
2. $\beta = \bar{\alpha}$.

For the first part we need to check that for all $f\colon n \longrightarrow K \in \mathbb{F}$ the following naturality square commutes

$$\begin{tikzcd} Fn \arrow[r, "\alpha_n"] \arrow[d, "Ff"'] & Gn \arrow[d, "Gf"] \\ Fk \arrow[r, "\alpha_k"'] & Gk \end{tikzcd}$$

Now by naturality of β we have

$$\begin{tikzcd} \mathbf{Set}(Fn, Fn) \arrow[r, "\alpha_n \circ \_"] \arrow[d, "Ff \circ \_"'] & \mathbf{Set}(Fn, Gn) \arrow[d, "Gf \circ \_"] \\ \mathbf{Set}(Fn, Fk) \arrow[r, "\alpha_k \circ \_"'] & \mathbf{Set}(Fn, Gk) \end{tikzcd}$$

so starting with the identity in the top left we have

$$\begin{tikzcd} 1_{Fn} \arrow[r, mapsto] \arrow[dd, mapsto] & \alpha_n \arrow[d, mapsto] \\ & Gf \circ \alpha_n \arrow[d, equal] \\ Ff \arrow[r, mapsto] & \alpha_k \circ Ff. \end{tikzcd}$$

Now we need to show that

$$\beta_{n,m}\colon \bar{F}(n,m) \longrightarrow \bar{G}(n,m)$$

is $\alpha_n \circ \_$, that is, $\beta_{n,1} \circ \_$. Now given $f\colon m \longrightarrow Fn$ and $i\colon 1 \longrightarrow m$ we have

$$\begin{tikzcd} \mathbf{Set}(m, Fn) \arrow[r, "\beta_{m,n}"] \arrow[d, "\_ \circ i"'] & \mathbf{Set}(m, Gn) \arrow[d, "\_ \circ i"] \\ \mathbf{Set}(1, Fn) \arrow[r, "\beta_{n,1} = \alpha_n \circ \_"'] & \mathbf{Set}(1, Gn) \end{tikzcd}$$

$$\begin{tikzcd}
f \arrow[r, mapsto] \arrow[dd, mapsto] & \beta_{m,n}(f) \arrow[d, mapsto] \\
& (\beta_{m,n}(f))(i) \arrow[d, equal] \\
F(i) \arrow[r, mapsto] & (\alpha_n \circ f)(i).
\end{tikzcd}$$

This is true for all $i \in m$, so $\beta_{m,n}(f)$ and $\alpha_n \circ f$ agree everywhere, hence $\beta = \bar{\alpha}$ as required and the functor ϕ is indeed full. □

Proposition 7.2. *The functor ϕ factors through* $\mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}}, \mathbb{F}^{\mathrm{op}})$

Proof. We use the definition of $\mathbf{Prof}(\mathbf{Mon})$ as $\mathbf{Mod}(\mathbf{Span}(\mathbf{Mon}))^{\mathrm{op}}$, and $\mathbf{Prof}$ as $\mathbf{Mod}(\mathbf{Span})^{\mathrm{op}}$. We write the underlying span of $\mathbb{F}^{\mathrm{op}}$ as

$$\begin{tikzcd}
& \mathbb{F}_1 \arrow[dl, "t"'] \arrow[dr, "s"] & \\
\mathbb{F}_0 & & \mathbb{F}_0
\end{tikzcd}$$

Then the underlying span of $\bar{F}$ as an $\mathbb{F}^{\mathrm{op}}$-bimodule is

$$\begin{tikzcd}
& A = \coprod_{m,n} \mathbf{Set}(m, Fn) \arrow[dl] \arrow[dr] & \\
\mathbb{F}_0 & & \mathbb{F}_0
\end{tikzcd}$$

We need to put a monoid structure on A such that the left and right $\mathbb{F}$-actions respect this. Note that the monoid structure in $\mathbb{F}_0$ is given by addition.

So given

$$f_1 \colon m_1 \longrightarrow Fn_1$$
$$f_2 \colon m_2 \longrightarrow Fn_2$$

we need a function

$$f_1 \oplus f_2 \colon m_1 + m_2 \longrightarrow F(n_1 + n_2).$$

Now by coproduct in **Set** we certainly have

$$m_1 + m_2 \xrightarrow{f_1 + f_2} Fn_1 + Fn_2 \xrightarrow{\text{canonical}} F(n_1 + n_2)$$

and we call this $f_1 \oplus f_2$. We also need $e \colon 0 \longrightarrow F0$ such that

$$f \oplus 0 = 0 \oplus f = f.$$

This is the unique map. Then $f \oplus 0$ is the following map:

$$m = m + 0 \xrightarrow{f+!} Fn + F0 \xrightarrow{F} (n + 0) = Fn$$

which is the same as f by a straightforward diagram chase.

Now we must check actions. These are given by pre- and post-composition. For the left action, given $k \xrightarrow{f} m$ in **Set** we have

$$\mathbf{Set}(m, Fn) \xrightarrow{\_\circ f} \mathbf{Set}(k, Fn)$$

and we need to check that

$$(g_1 \oplus g_2) \circ (f_1 + f_2) = (g_1 \circ f_1) \oplus (g_2 \circ f_2).$$

$$\begin{tikzcd}
 & F(n_1+n_2) & \\
 & Fn_1+Fn_2 \arrow[u, dashed] & \\
Fn_1 \arrow[ur] \arrow[uur, bend left] & & Fn_2 \arrow[ul] \arrow[uul, bend right] \\
 & m_1'+m_2' \arrow[uu, dashed, "g_1+g_2"'] & \\
m_1' \arrow[uu, "g_1"] \arrow[ur] & & m_2' \arrow[uu, "g_2"'] \arrow[ul] \\
 & m_1+m_2 \arrow[uu, dashed, "f_1+f_2"'] & \\
m_1 \arrow[uu, "f_1"] \arrow[ur] & & m_2 \arrow[uu, "f_2"'] \arrow[ul]
\end{tikzcd}$$

The left and right hand sides of the equation we want then just correspond to the middle dotted composite of this diagram associated either way round, so the result follows by associativity.

For the right action, given $n \xrightarrow{f} k$ in **Set** we have

$$\mathbf{Set}(m, Fn) \xrightarrow{Ff\circ\_} \mathbf{Set}(m, Fk)$$

and we need to check that

$$F(f_1 + f_2) \circ (g_1 \oplus g_2) = (Ff_1 \circ g_1) \oplus (Ff_2 \circ g_2).$$

The result then follows by a straightforward diagram chase involving diagrams similar to the previous one.

Now we must show that $\bar{\alpha}$ is a monoid map as

$$\coprod_{m,n} \mathbf{Set}(m, Fn) \longrightarrow \coprod_{m,n} \mathbf{Set}(m, Gn).$$

So we need to show that given

$$f_1 \colon m_1 \longrightarrow Fn_1$$
$$f_2 \colon m_2 \longrightarrow Fn_2$$

we have

$$\alpha_{n_1+n_2} \circ (f_1 \oplus f_2) = (\alpha_{n_1} \circ f_1) \oplus (\alpha_{n_2} \circ f_2).$$

This follows from a straightforward diagram chase and naturality of α. $\square$

Corollary 7.3. *It follows immediately that*

$$\mathbf{CAT}_f(\mathbf{Set},\mathbf{Set}) \xrightarrow{\psi} \mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}},\mathbb{F}^{\mathrm{op}})$$

is full as well as faithful, and likewise the forgetful functor

$$\mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}},\mathbb{F}^{\mathrm{op}}) \xrightarrow{U^{\psi}} \mathbf{Prof}^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}},\mathbb{F}^{\mathrm{op}})$$

is also full and faithful.

Proof. Follows from ϕ being full (Proposition 7.1). □

Corollary 7.4.

1. *When F is a monoid in $\mathbf{CAT}_f(\mathbf{Set},\mathbf{Set})$ (i.e. a finitary monad on $\mathbf{Set}$), ψF is a monad in $\mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}$ given by an identity-on-objects functor*

$$\alpha\colon \mathbb{F}^{\mathrm{op}} \longrightarrow A$$

where the monoidal structure of A is given by products. Conversely any such monad in $\mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}$ arises in this way.

2. *When F is a monoid in $\mathbf{CAT}_f(\mathbf{Set},\mathbf{Set})$, ϕF is a monad in $\mathbf{Prof}^{\mathrm{op}}$ given by an identity-on-objects functor*

$$\alpha\colon \mathbb{F}^{\mathrm{op}} \longrightarrow A$$

where A has finite products and α preserves finite products. Conversely any such monad in $\mathbf{Prof}^{\mathrm{op}}$ arises in this way.

Remark 7.5. Note that this is not much more than the standard correspondence between finitary monads and Lawvere theories.

Proof.
Regarding $\phi F = \bar{F}$ as a category, $\bar{F}(n,m) = \mathbf{Set}(m, Fn)$. We need to check that $n+k$ is the categorical product in $\bar{F}$, that is, there is a natural isomorphism

$$\bar{F}(p,n) \times \bar{F}(p,k) \cong \bar{F}(p, n+k)$$

that is

$$\mathbf{Set}(n, Fp) \times \mathbf{Set}(k, Fp) \cong \mathbf{Set}(n+k, Fp)$$

which is true by definition of coproduct in $\mathbf{Set}$. This proves both parts. □

Corollary 7.6. *Let A and B be Lawvere theories expressed according to any of Definitions 5.5 "PROF", 4.12 "FS" or 6.8 "KLEISLI", and let $\sigma\colon AB \longrightarrow BA$ be a distributive law of A over B according to the same definition. Then the composite monad BA is also a Lawvere theory.*

Proof.
For monads in $\mathbf{Prof}^{\mathrm{op}}$ we know can write A and B as ϕS and ϕT for some finitary monads S and T by Corollary 7.4. Then by fullness of ϕ the distributive

law σ must be of the form $\bar{\lambda}$ for some distributive law of S over T. Thus the composite BA is isomorphic to ϕTS thus is a Lawvere theory. The result for factorisation systems immediately follows, and that for monads in $\mathbf{Prof}(\mathbf{Mon})^{\mathrm{op}}$ follows in the same way $\square$

Although we have now completed the equivalences, we include one further characterisation as we find it illuminating. It is well known that there are two canonical identity-on-objects functors relating **CAT** and **PROF**. Given a functor $C \xrightarrow{F} D$ in **CAT** the two functors act as follows.

1. Covariant: $C \overset{F_*}{\longrightarrow\!\!\!\!\!\!\!\!+\;\;\,} D$ in **PROF** defined by $F_*(d,c) = D(d,Fc)$. This is the canonical free functor if we regard **PROF** as the Kleisli bicategory for the presheaf monad.

2. Contravariant: $D \overset{F^*}{\longrightarrow\!\!\!\!\!\!\!\!+\;\;\,} C$ defined by $F^*(c,d) = F(Fc,d)$.

Thus given a monad $\mathbf{Set} \xrightarrow{T} \mathbf{Set}$ we get a monad $\mathbf{Set} \overset{T_*}{\longrightarrow\!\!\!\!\!\!\!\!+\;\;\,} \mathbf{Set}$ in **PROF** and this could be regarded as an algebraic theory typed in **Set**. However if we have a finitary monad we can restrict our types to the small category $\mathbb{F}$ via a chosen embedding

$$\mathbb{F} \xrightarrow{I} \mathbf{Set}.$$

Then we can define a functor

$$\begin{array}{ccc} \mathbf{CAT}_f(\mathbf{Set},\mathbf{Set}) & \longrightarrow & \mathbf{Prof}(\mathbb{F},\mathbb{F}) \\ \mathbf{Set} \xrightarrow{F} \mathbf{Set} & \longmapsto & \mathbb{F} \overset{I_*}{\longrightarrow\!\!\!\!\!\!\!\!+\;\;\,} \mathbf{Set} \overset{F_*}{\longrightarrow\!\!\!\!\!\!\!\!+\;\;\,} \mathbf{Set} \overset{I^*}{\longrightarrow\!\!\!\!\!\!\!\!+\;\;\,} \mathbb{F}. \end{array}$$

The following proposition shows that on monads this gives us the (opposite of) the associated Lawvere theory.

Proposition 7.7. *The above composite gives the profunctor*

$$(k,n) \longmapsto \mathbf{Set}(k,Fn).$$

Proof. This is a straightforward coend calculation, using the fact that $F_*I_* = (FI)_*$:

$$\begin{aligned} (k,n) \quad &\mapsto \quad \int^{X\in\mathbf{Set}} I^*(k,X) \times (FI)_*(X,n) \\ &= \quad \int^{X\in\mathbf{Set}} \mathbf{Set}(k,X) \times \mathbf{Set}(X,Fn) \\ &= \quad \mathbf{Set}(k,Fn) \qquad\qquad \text{by density.} \end{aligned}$$

Finally we can regard this as $\mathbb{F}^{\mathrm{op}} \longrightarrow\!\!\!\!\!\!\!\!+\;\;\, \mathbb{F}^{\mathrm{op}}$ by taking it to be in $\mathbf{Prof}^{\mathrm{op}}$ via the standard duality. $\square$

Hence we have directly constructed the functor

$$\mathbf{CAT}_f(\mathbf{Set},\mathbf{Set}) \longrightarrow \mathbf{Prof}^{\mathrm{op}}(\mathbb{F}^{\mathrm{op}},\mathbb{F}^{\mathrm{op}})$$

constructed previously as the composite via $\mathbf{Prof}_{\mathcal{P}}$, and this gives another explanation of the (slightly annoying) presence of the "op".

Furthermore, that this functor is monoidal follows neatly from the finitary conditions as follows. We need to check that, given finitary functors

$$\mathbf{Set} \xrightarrow{F} \mathbf{Set} \xrightarrow{G} \mathbf{Set}$$

the composite in **Prof**

$$\mathbb{F} \xrightarrow{I_*} \mathbf{Set} \xrightarrow{F_*} \mathbf{Set} \xrightarrow{G_*} \mathbf{Set} \xrightarrow{I^*} \mathbb{F}$$

is isomorphic to

$$\mathbb{F} \xrightarrow{I_*} \mathbf{Set} \xrightarrow{F_*} \mathbf{Set} \xrightarrow{I^*} \mathbb{F} \xrightarrow{I_*} \mathbf{Set} \xrightarrow{G_*} \mathbf{Set} \xrightarrow{I^*} \mathbb{F}.$$

In fact G being finitary gives us that

$$\mathbf{Set} \xrightarrow{G_*} \mathbf{Set} \xrightarrow{I^*} \mathbb{F}$$

is isomorphic to

$$\mathbf{Set} \xrightarrow{I^*} \mathbb{F} \xrightarrow{I_*} \mathbf{Set} \xrightarrow{G_*} \mathbf{Set} \xrightarrow{I^*} \mathbb{F}.$$

We simply calculate the coends. The first gives

$$\begin{aligned} (k, X) \quad \mapsto \quad & \int^{A \in \mathbf{Set}} G_*(A, X) \times I^*(k, A) \\ = \quad & \int^{A \in \mathbf{Set}} \mathbf{Set}(A, GX) \times \mathbf{Set}(k, A) \\ = \quad & \mathbf{Set}(k, GX) \qquad\qquad \text{by density.} \end{aligned}$$

For the second composite we have

$$\begin{aligned} (k, X) \quad \mapsto \quad & \int^{n \in \mathbb{F}} \mathbf{Set}(k, Gn) \times \mathbf{Set}(n, X) \\ = \quad & \prod_k \int^{n \in \mathbb{F}} Gn \times X^n \qquad\qquad \text{as coends commute with finite products} \\ = \quad & \prod_k GX \qquad\qquad \text{as } G \text{ is finitary} \\ = \quad & \mathbf{Set}(k, GX) \end{aligned}$$

as required.

8 Future work

This new theory of distributive laws for Lawvere theories, with its four different viewpoints, opens up various possibilities for further study. We conclude by briefly mentioning a few.

- We could seek more concrete ways of expressing distributive laws over $\mathbb{F}^{\mathrm{op}}$ using the (quite special) properties of $\mathbb{F}^{\mathrm{op}}$. We could seek "canonical forms" for operations in the composite theory.
- We could study the question of when an algebraic theory can be decomposed into simpler ones, and when it is "irreducible",
- We could further study iterated distibutive laws in the context of Lawvere theories.
- We could extend the theory to any of the generalised versions of Lawvere theory.

References

[1] Andrei Akhvlediani. Composing Lawvere theories, 2010. Talk at CT2010.

[2] J. Beck. Distributive laws. *Lecture Notes in Mathematics*, 80:119–140, 1969.

[3] Eugenia Cheng. Iterated distributive laws. *Mathematical Proceedings of the Cambridge Philosophical Society*, 150(3):459–487, 2011. Also E-print `0710.1120`.

[4] Eugenia Cheng, Martin Hyland, and John Power. Pseudo-distributive laws. *Electronic Notes in Theoretical Computer Science*, 83, 2004.

[5] Martin Hyland. On distributive laws, 2010. Talk at Categories, Logic and Physics VI.

[6] Martin Hyland and John Power. The category theoretic understanding of universal algebra: Lawvere theories and monads. *Electronic Notes in Theoretical Computer Science*, 172:437–458, 2007.

[7] S. Lack and J. Rosický. Notions of Lawvere theory. *Applied Categorical Structures*, 175(1):243–265, 2011. Special volume celebrating the 70th birthday of Professor Max Kelly.

[8] Stephen Lack. Composing PROPs. *Theory and Applications of Categories*, 13:147–163, 2004.

[9] F. W. Lawvere. *Functional semantics of algebraic theories.* PhD thesis, Columbia University, 1963. Also available as TAC Reprint 5, `http://tac.mta.ca/tac/reprints/articles/5/tr5abs.html`.

[10] F. E. J. Linton. Some aspects of equational theories. In *Proc. Conf. on Categorical Algebra at La Jolla*, pages 84–95, 1966.

[11] John Power. Enriched Lawvere theories. *Theory and Applications of Categories*, 6(7):83–93, 1999.

[12] Robert Rosebrugh and Richard Wood. Distributive laws and factorization. *J. Pure Appl. Algebra*, 175:327–353, 2002.

[13] Ross Street. The formal theory of monads. *Journal of Pure and Applied Algebra*, 2:149–168, 1972.